Package ‘ShiVa’

July 22, 2025

Type Package

Title Detection of Evolutionary Shifts in Both Optimal Value and Variance

Version 1.0.1

Description Implements statistical methods for detecting evolutionary shifts in both the optimal trait value (mean) and evolutionary diffusion variance. The method uses an L1-penalized optimization framework to identify branches where shifts occur, and the shift magnitudes. It also supports the inclusion of measurement error. For more details, see Zhang, Ho, and Kenney (2023) <doi:10.48550/arXiv.2312.17480>.

License GPL (>= 3)

Encoding UTF-8

Depends R (>= 3.6.0)

Imports glmnet, psych, ape, phylolm, MASS, igraph

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.3.2

NeedsCompilation no

Author Wensha Zhang [aut, cre],
Lam Si Tung Ho [aut],
Toby Kenney [aut]

Maintainer Wensha Zhang `<wn209685@dal.ca>`

Repository CRAN

Date/Publication 2025-07-22 10:10:11 UTC

Contents

`backward_correction` *Backward Selection for OU Model Shift Correction*

Description

Performs backward stepwise selection on a given set of candidate shifts in optimal trait values (mean) and evolutionary variance under an Ornstein-Uhlenbeck (OU) model. This function iteratively removes individual shifts to improve model fit based on a specified selection criterion.

Usage

```
backward_correction(
  tree,
  Y,
  alpha,
  shifts_mean,
  shifts_var,
  criterion = "BIC",
  original_model = NULL,
  measurement_error = FALSE,
  max.num.shifts = Inf
)
```

Arguments

`tree`	A phylogenetic tree of class `phylo`.
`Y`	A numeric vector of trait values corresponding to the tips of the tree.
`alpha`	A non-negative numeric value specifying the strength of selection in the OU process.
`shifts_mean`	A vector of branch indices with candidate shifts in optimal trait values.
`shifts_var`	A vector of branch indices with candidate shifts in evolutionary variance.
`criterion`	A model selection criterion to guide backward elimination. Options include `"BIC"`, `"mBIC"`, or `"pBIC"`. Default is `"BIC"`.
`original_model`	(Optional) A previously fitted OU model returned by `fit_OU_mean_var`. If `NULL`, the model is refit using the provided shifts.

`measurement_error`	Logical. If `TRUE`, the model accounts for measurement error by estimating an additional variance term. Default is `FALSE`.
`max.num.shifts`	An integer specifying the maximum number of total shifts (mean and variance combined) allowed in the model. Default is `Inf`.

Value

A fitted OU model object (a list), as returned by `fit_OU_mean_var`, with a potentially reduced set of shifts that minimizes the specified criterion.

`fit_OU_mean_var` *Fit OU Model with Shifts in Mean and Variance*

Description

Fits an Ornstein-Uhlenbeck (OU) model with user-specified shifts in both optimal trait values (mean) and evolutionary variance along a phylogeny. The method uses numerical optimization to estimate shift magnitudes, base variance, and intercept, and can optionally incorporate measurement error in trait values.

Usage

```
fit_OU_mean_var(
  tree,
  Y,
  alpha,
  shifts_mean,
  shifts_var,
  max.steps = 1000,
  t = 0.01,
  thres = 0.01,
  measurement_error = FALSE,
  max.num.shifts = Inf
)
```

Arguments

`tree`	A phylogenetic tree of class `phylo`.
`Y`	A numeric vector of continuous trait values at the tips of the tree.
`alpha`	A non-negative numeric value specifying the strength of selection in the OU process.
`shifts_mean`	An integer vector indicating the indices of branches where shifts in the optimal trait value occur.
`shifts_var`	An integer vector indicating the indices of branches where shifts in evolutionary variance occur.

`max.steps`	Maximum number of optimization steps. Default is 1000.
`t`	Step size for the optimizer. Default is 0.01.
`thres`	Convergence threshold for change in log-likelihood. Default is 0.01.
`measurement_error`	Logical. If `TRUE`, a separate measurement error variance is estimated and added to the diagonal of the covariance matrix.
`max.num.shifts`	Maximum allowed number of shifts (combined for mean and variance). Default is `Inf`.

Value

A list containing:

`tree`	The phylogenetic tree.
`Y`	The trait values.
`shifts_mean`	Indices of branches with non-zero shifts in optimal trait value.
`shifts_var`	Indices of branches with non-zero shifts in evolutionary variance.
`beta`	Estimated shift magnitudes for optima values.
`gamma`	Estimated shift magnitudes for variance.
`sigma2`	Estimated base evolutionary variance.
`b0`	Estimated intercept (ancestral trait value).
`sigma2_error`	Estimated measurement error variance (only returned if `measurement_error = TRUE`).
`loglik`	Log-likelihood of the fitted model.
`BIC`	BBIC for model selection.
`mBIC`	mBIC for accounting shift sparsity and shared support.
`pBIC`	pBIC incorporating determinant of projected design matrix.
`fitted.values`	Fitted trait values based on the estimated model.
`Sigma`	Estimated trait covariance matrix under the fitted model.

`generate_design_matrix`

Generate Design Matrix

Description

Constructs a design matrix for a given phylogenetic tree, used in Ornstein-Uhlenbeck modeling.

Usage

```
generate_design_matrix(tree, type = "simpX", alpha = 0)
```

Arguments

`tree`	A phylogenetic tree of class `phylo`.
`type`	A character string specifying the type of design matrix to generate. Options are `"simpX"` or `"orgX"`.
`alpha`	The selection strength parameter (only used when `type = "orgX"`).

Value

A design matrix `X`, where each row corresponds to a tip and each column to an edge in the tree.

`get_mean_var_shifts` *Estimate Shifts in Optimal Trait Values and Variance*

Description

Estimates shifts in both the optimal trait values (mean) and evolutionary variance along a phylogeny under an Ornstein-Uhlenbeck (OU) process, using an ℓ_1-penalized optimization procedure. Optionally accounts for measurement error in the observed trait data.

Usage

```
get_mean_var_shifts(
  Y,
  tree,
  alpha,
  lambda1,
  lambda2,
  max.steps = 1000,
  t = 0.01,
  penalty = "L1",
  thres = 0.01,
  sigma2 = NULL,
  measurement_error = FALSE
)
```

Arguments

`Y`	A numeric vector of continuous trait values for the species at the tips of the tree.
`tree`	A phylogenetic tree of class `phylo`.
`alpha`	The selection strength parameter in the OU process.
`lambda1`	Non-negative penalty for β (shifts in optimal trait values).
`lambda2`	Non-negative penalty for γ (shifts in evolutionary variance).
`max.steps`	Maximum number of optimization steps. Default is 1000.
`t`	Step size for the gradient-based updates. Default is 0.01.

`penalty`	Type of penalty to apply. Options are `"L1"` (default) or `"None"`.
`thres`	Convergence threshold for the change in loss between steps. Default is 0.01.
`sigma2`	Optional initial value for the base evolutionary variance. If `NULL`, it is initialized to 1.
`measurement_error`	Logical. If `TRUE`, the method estimates additional measurement error variance.

Value

A list containing:

`shifts_mean`	Indices of branches with detected shifts in optimal trait values ($\beta \neq 0$).
`shifts_var`	Indices of branches with detected shifts in evolutionary variance ($\gamma \neq 0$).
`beta`	Estimated shift coefficients for optimal trait values.
`gamma`	Estimated shift coefficients for evolutionary variance.
`sigma2`	Estimated base variance (σ^2) of the OU process.
`b0`	Estimated intercept (root state).
`sigma2_error`	Estimated measurement error variance (only returned if `measurement_error = TRUE`).

`get_mean_var_shifts_model_selection`

Model Selection for OU Shifts in Optimal value and Variance

Description

Performs model selection to estimate the locations and magnitudes of evolutionary shifts in optimal trait values (mean) and diffusion variance under an Ornstein-Uhlenbeck (OU) process. This function searches across user-defined grids of shrinkage parameters for both types of shifts, uses cross-validation for selecting `lambda1`, and applies backward correction to refine top candidate models.

Usage

```
get_mean_var_shifts_model_selection(
  Y,
  tree,
  alpha,
  t = 0.01,
  lambda1_list = NULL,
  lambda2_list = exp(1:10 * 0.4 - 6),
  criterion = "BIC",
  max.steps = 300,
  nfolds = 8,
  top_k = 10,
```

```
  measurement_error = FALSE,
  lambda.type = "lambda.1se",
  max.num.shifts = Inf,
  verbose = TRUE
)
```

Arguments

`Y`	A numeric vector of trait values for the species at the tips of the phylogenetic tree.
`tree`	A phylogenetic tree of class `phylo`.
`alpha`	A non-negative numeric value representing the selection strength in the OU process.
`t`	Step size for iterative optimization. Default is 0.01.
`lambda1_list`	A numeric vector of candidate λ_1 values controlling shrinkage for shifts in optimal values.
`lambda2_list`	A numeric vector of candidate λ_2 values controlling shrinkage for shifts in variance. Default is exp(1:10*0.4-6)
`criterion`	Model selection criterion to optimize. Options include `"BIC"`, `"mBIC"`, or `"pBIC"`. Default is `"BIC"`.
`max.steps`	Maximum number of optimization steps. Default is 300.
`nfolds`	Number of cross-validation folds for tuning `lambda1`. Default is 8.
`top_k`	Number of top candidate models (ranked by criterion) to further refine using backward correction. Default is 10.
`measurement_error`	Logical. If `TRUE`, estimates a separate measurement error variance component. Default is `FALSE`.
`lambda.type`	A character string specifying the cross-validation rule used to select `lambda1` from `lambda1_list`. Options are `"lambda.min"` (minimum CV error) and `"lambda.1se"` (1-SE rule, higher penalty). Default is `"lambda.1se"`.
`max.num.shifts`	An integer specifying the maximum number of allowed shifts (combined across mean and variance). Default is `Inf`.
`verbose`	Logical. If `TRUE`, prints progress messages to the console. Default is `TRUE`.

Value

A list containing:

`best_model`	The final selected OU model object, with estimated shifts and parameters.
`score_summary`	A data frame summarizing the model selection results, including pre- and post-correction scores and shift locations.

`OU.vcv` *OU.vcv*

Description

generate covariance matrix for OU process

Usage

```
OU.vcv(tree, alpha)
```

Arguments

`tree`	phylogenetic tree
`alpha`	selective force

Value

`V`	covariance matrix

`plot.ShiftModel` *Plot Method for ShiftModel Objects*

Description

Plots a phylogenetic tree with trait values at the tips and highlights branches with detected shifts in optimal value (mean) and variance.

Usage

```
## S3 method for class 'ShiftModel'
plot(x, title = "", ...)
```

Arguments

`x`	An object of class `ShiftModel`, typically returned by `fit_OU_mean_var()` or `ShiVa()`.
`title`	A character string specifying the plot title.
`...`	Additional arguments passed to `plot.phylo()`.

Value

No return value. This function is called for its side effect: a plotted tree.

`print.summary.ShiftModel` *Print Method for Summary of ShiftModel*

Description

Prints a formatted summary of a fitted `ShiftModel`, including the estimated parameters (`alpha`, `sigma2`, log-likelihood, and BIC), as well as details of detected shifts in the optimal trait value (mean) and variance. This method is typically called on the result of `summary()` applied to an object of class `ShiftModel`.

Usage

```
## S3 method for class 'summary.ShiftModel'
print(x, ...)
```

Arguments

`x`	An object of class `summary.ShiftModel`, usually returned by `summary(model)` where `model` is a `ShiftModel`.
`...`	Additional arguments (currently ignored).

Value

No return value. Called for its side effect: printing summary information to the console.

`ShiVa` *ShiVa: Automatic Shift Detection in Mean and Variance*

Description

Performs automatic detection of evolutionary shifts in both optimal trait values (mean) and diffusion variance under an Ornstein-Uhlenbeck (OU) process. This function serves as a wrapper for `get_mean_var_shifts_model_selection`, with the added ability to automatically estimate the selection strength parameter `alpha` if not provided.

Usage

```
ShiVa(
  Y,
  tree,
  alpha = NULL,
  t = 0.01,
  lambda1_list = NULL,
  lambda2_list = exp(1:10 * 0.4 - 6),
```

```
  criterion = "BIC",
  max.steps = 300,
  nfolds = 8,
  top_k = 10,
  measurement_error = FALSE,
  lambda.type = "lambda.1se",
  max.num.shifts = Inf
)
```

Arguments

`Y`	A numeric vector of trait values at the tips of the phylogenetic tree.
`tree`	A phylogenetic tree of class `phylo`.
`alpha`	(Optional) A non-negative numeric value specifying the OU selection strength. If `NULL`, it is estimated via maximum likelihood using `phylolm()`.
`t`	Step size for optimization. Default is 0.01.
`lambda1_list`	A numeric vector of candidate λ_1 values for penalizing mean shifts.
`lambda2_list`	A numeric vector of candidate λ_2 values for penalizing variance shifts. Default is `exp(1:10 * 0.4 - 6)`.
`criterion`	Model selection criterion to use. Options are `"BIC"`, `"mBIC"`, or `"pBIC"`. Default is `"BIC"`.
`max.steps`	Maximum number of optimization steps. Default is 300.
`nfolds`	Number of folds for cross-validation in tuning `lambda1`. Default is 8.
`top_k`	Number of top candidate models (based on criterion) to refine using backward correction. Default is 10.
`measurement_error`	Logical. If `TRUE`, estimates a measurement error variance term. Default is `FALSE`.
`lambda.type`	Cross-validation rule for selecting `lambda1`. Options are `"lambda.min"` or `"lambda.1se"`. Default is `"lambda.1se"`.
`max.num.shifts`	Maximum number of allowed shifts (in both mean and variance). Default is `Inf`.

Value

A list with the same structure as `get_mean_var_shifts_model_selection`:

`best_model`	The final selected OU model object.
`score_summary`	A data frame summarizing candidate models and backward-corrected scores.

`soft_thresholding` *Soft Thresholding*

Description

Applies the soft thresholding operation to a numeric input, commonly used in Lasso and sparse modeling to induce shrinkage and sparsity.

Usage

```
soft_thresholding(z, lambda)
```

Arguments

`z`	A numeric value to be thresholded.
`lambda`	A non-negative numeric value indicating the threshold level (degree of shrinkage).

Value

A numeric value after applying soft thresholding: $\text{sign}(z) \cdot \max(|z| - \lambda, 0)$.

`summary.ShiftModel` *Summary of a ShiVa Shift Model*

Description

Generate a summary of a fitted ShiVa model object, including estimated parameters and details of detected shifts in optimal trait value (mean) and evolutionary variance.

Usage

```
## S3 method for class 'ShiftModel'
summary(object, ...)
```

Arguments

`object`	An object of class `ShiftModel`, typically returned by functions such as `fit_OU_mean_var()`, or extracted as `best_model` from `get_mean_var_shifts_model_selection()` or `ShiVa()`.
`...`	Additional arguments (currently unused).

Value

An object of class `summary.ShiftModel`, which includes:

- `alpha`: Estimated selection strength parameter.
- `sigma2`: Estimated diffusion variance.
- `loglik`: Log-likelihood of the fitted model.
- `BIC`: Bayesian Information Criterion.
- `mean_shifts`: A data frame of branches with shifts in optimal value.
- `var_shifts`: A data frame of branches with shifts in evolutionary variance.

`update_step_gamma` *Update Step for Gamma*

Description

Performs one update step for the `gamma_k` parameter in an iterative optimization routine, potentially applying L1 shrinkage.

Usage

```
update_step_gamma(gamma_k, X_k, Sigma, r, lambda2, t, penalty, V, q_k)
```

Arguments

`gamma_k`	Current value of the `gamma_k` parameter to be updated.
`X_k`	The kth column of the design matrix `X`.
`Sigma`	Current covariance matrix.
`r`	Residual vector, typically `Y - X %*% beta`.
`lambda2`	Non-negative tuning parameter controlling the degree of L1 shrinkage applied to `gamma_k`.
`t`	Step size for the gradient update.
`penalty`	Penalty type; either `"L1"` for soft thresholding or `"None"` for unpenalized updates.
`V`	Baseline covariance matrix when $\sigma^2 = 1$.
`q_k`	The kth element of the design vector `q`, used in the update.

Value

A list containing:

`gamma_k`	The updated value of `gamma_k`.
`Sigma`	The updated covariance matrix `Sigma`.

Index